REMARKS

Reconsideration of the application is respectfully requested for the following reasons:

1. Rejection of Claims 26-33 and 42 Under 35 USC §101

This rejection has been addressed by amending claim 26 to positively recite execution of the operations *f* on the semiconductor chip and, before combining the output data determined by execution of the one or more operations with the auxiliary function value, retrieving the auxiliary function value from the memory on the chip.

Clearly, execution of operations on a chip and retrieval of a function from a memory on the chip must be performed by a "machine" and are not merely abstract mathematical concepts. Furthermore, both the execution and retrieval steps must be performed after data falsification in order to carry out the combining step, *i.e.*, the solution cannot be obtained without performing these machine-implemented steps, and therefore the steps are not mere post or extra-solution activities.

Withdrawal of the rejection of claims 26-33 and 42 under 35 USC §101 is accordingly respectfully requested.

2. Rejection of Claims 26-33 and 42 Under 35 USC §103(a) in view of U.S. Patent Publication No. 2002/0124178 (Kocher) and U.S. Patent No. 5,655,023 (Cordery)

This rejection is again respectfully traversed on the grounds that the Kocher publication and the Cordery patent fail to disclose or suggest a data carrier, in which:

- falsified input data is used to prevent signals caused by execution of operations on a semiconductor chip, and
- the falsified data is compensated for by combining the output with an auxiliary function value that is **pre-stored and retrieved from a memory on the chip** ***rather than being generated during falsification of the input data***,
- the auxiliary function value having been previously determined by the execution of the one or more operations with the auxiliary data as input data in safe surroundings and stored along with the auxiliary data.

as recited in claim 26. Instead, the Kocher publication teaches that auxiliary functions may be generated **during** falsification of the input data, while the Cordery patent merely teaches pre-storage of keys and other secret data in general, without any including any teachings that would have suggested that the ordinary artisan should modify the method of Kocher by eliminating the step of generating auxiliary function values during data falsification, or providing any reason for the modification.

It appears from the Examiner's response to Applicant's arguments, on pages 2-8 of the Official Action, that the Examiner has misunderstood the arguments made in the last response. Contrary to the statement in the first complete paragraph on page 4 of the Official Action, the Applicant is not merely arguing that Cordery is "**nonanalogous art**" because it is not from the same field of endeavor,[1] nor is Applicant arguing that Cordery's teachings cannot be **bodily incorporated** into Kocher's method, or that there must be an **express suggestion** in order to make the combination. Instead, the Applicant's main argument is that, **based on what is actually taught in the references**, one of ordinary skill in the art would not have found it obvious to combine those teachings.

According to the Examiner, Cordery's teaching of a postal meter that includes a removable token for storing secret keys would have cause the ordinary artisan to modify Kocher's method differential power analysis method by eliminating the auxiliary function value generation step and instead use a pre-stored auxiliary function value, and that any argument to the contrary is "largely conjecture" (see the second paragraph on page 7 of the Official Action). This analysis by the Examiner makes no sense. Kocher's system already includes a memory for storing a key, and yet Kocher still generates the auxiliary function value during data falsification rather than previously in safe surroundings. **By what logic would one of ordinary skill in the art decide that Cordery's general key storage arrangement 104 suggests modification of the power-analysis-preventing algorithm, when Kocher already includes a key storage**

[1] According to the Examiner, all data protection, from invisible ink to quantum scrambling is from the same field of endeavor and therefore combinable. Even assuming that this is true, however, one must still consider the specific nature of the teachings in the references in order to determine whether the combination is obvious–such as whether one of ordinary skill in the art would have a teaching of security token for a postage reader (Cordery) to be applicable to the problem of differential power analysis of radiation emitted by a processor on a data card (Kocher), especially when the security token of Cordery merely holds secret data and does not perform any processing operations, while the processor of Cordery is assumed to be invulnerable to differential power analysis and does not require any sort of auxiliary function values (at least no such auxiliary function values are taught).

arrangement 290 and does not use it to modify the power-analysis-preventing algorithm?
The fact that Kocher teaches a specific power analysis method is not "**conjecture**," nor is the teaching in Kocher of auxiliary function value generation during data falsification and, at the same time storage of keys other than the auxiliary function value in a memory. Furthermore, the fact that Cordery merely teaches storage of keys in a memory and has nothing to do with power analysis is not "**conjecture**." To the contrary, the only "conjecture" involved in this case is the conjecture that one of ordinary skill in the art would modify the specific auxiliary function value generation method taught by Kocher based on Cordery's general teachings of key storage.

Far from basing the response on "conjecture," the Applicant's arguments are based on a detailed analysis of the references. This is not to suggest that physical incorporation is required, but only an inquiry into what the references would have taught one of ordinary skill in the art. The Examiner is reminded that one of ordinary skill in the art, when considering the Kocher and Cordery references, would have considered all of the teachings in the references, in their proper context, when considering whether the teachings of Cordery of key storage, in a context that does not involve power analysis, would have suggested modification of an aspect of Kocher's power analysis method that has nothing to do with key storage. As explained in MPEP 2141.02, p. 2100-107, "[a] prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention" (emphasis in the original). The Examiner is reminded that one of ordinary skill in the art, when considering the Turk patent, would have considered all of the teachings in the Turk patent, in their proper context, when considering whether those teachings had any applicability to reel sensing devices. As explained in MPEP 2141.02, p. 2100-107, "[a] prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention" (emphasis in the original).

As explained in the previous response, when considering the teachings of Kocher as a whole, it can be seen that the array *dataIn* described in paragraphs [0068] to [0073] corresponds to the input data, the random bits *b* correspond to the auxiliary data, and the permutations defined by the arrays *table* and *perm* arguably correspond to the one or more operations of amended claim 26. The array *perm* represents an additional permutation that is computed randomly while computing the actual permutation defined by *table*. The output data *dataOut* representing a

permutation of *dataIn* according to *table* are in fact not affect by *perm* although *perm* is twice applied to the input data. As described in paragraphs [0067], [0068], the additional permutation *perm* is used to avoid processing the steps to compute the permutation *dataOut* in input order or in output order since both orders may lead to leakage of information, so that *dataOut* itself does not depend on the random array *perm*, but rather it is the order in which *dataOut*'s entries are computed that depends on the random array *perm*. The falsification of data, on the other hand, is described by adding a random bit *b* modulo 2 to the permuted input data *perm [i]*, and storing the falsified bit in an array *temp*. This is expressed in Kocher as *dataIn [p] ^ b = dataIn [perm[i]] ^ b (*where ^ is the modulo operation). Thus, the step of falsifying data in Kocher, *i.e.*, blinding a bit of the input data by adding a random bit b modulo 2 (as can be seen in the third for-loop of the pseudo-code in paragraph [0068]), **is only performed AFTER the step of performing the additional permutation *perm***, meaning that the corresponding auxiliary data and function value computation steps cannot be carried out before at least one of the operation steps (obtaining *perm*) is performed. Thus, an **essential** feature of the method of Kocher in order to prevent information leakage (paragraphs [0068] and [0069] and cannot be omitted without rendering the method of Kocher inoperative. **This is not a matter of "conjecture," as alleged by the Examiner, but what is actually taught by Kocher**. Furthermore, it is **NOT CONJECTURE** that Cordery does not contain a single teaching that has anything specific to do with data corresponding to dataIn or dataOut of Kocher. **Cordery merely teaches that keys and other secret data can be stored, which Kocher already does in memory 290, but cannot be reasonably said to suggest changing the order of the third FOR loop in paragraph [0068] of Kocher and the step of performing the additional permutation *perm*.**

According to the present invention, the input data are falsified by combination with auxiliary data before the execution of the one or more operations. According to Kocher, on the other hand, the step of falsifying the input data is performed only after one of the operations, namely the permutation *perm*, has already been performed on the input data. Since Cordery's general teachings of secret data pre-computation do not provide a way to avoid Kocher's requirement that the permutation *perm* be applied to the input data before blinding of the permuted input data in order to prevent data leakage, the proposed combination could not have resulted in the claimed invention. Interchanging the permutation and falsifying (blinding) operations would be contrary to the teachings of Kocher because then the order in which the

blinding steps would be executed would correspond to the standard input order (the steps would be performed in the order of index *i* from 0 to 63) with the above-mentioned risk of information leakage. Applying the permutation *perm* after the blinding steps would be useless and therefore not an obvious modification.

Nothing in the Cordery patent would have cause the ordinary artisan to ignore the teaching in Kocher that an appropriate unblinding vector is stored in the array *dataOut* and already computed together with the blinded input vector, *i.e.*, in the same for-loop defined by *dataOut ptable[p] := b*. Again, this conclusion is not based on conjecture. To the contrary, the Examiner's conclusion that one of ordinary skill in the art would have been caused by Cordery to ignore these teachings is pure conjecture. The Examiner will note that the left hand side equals *dataOut [table[perm[i]]*, *i.e.*, that the unblinding vector is determined by applying the permutations *perm* and *table* to the random vector *b* (the random bit *b* in step *i* of the for-loop being interpreted as an entry *b[i]* of a respective vector *b*). After the permutation defined by the array *table* is applied as the second of the one or more operations, to the falsified input data in the fourth for-loop, the appropriate compensation for the prior falsification of the input data follows by means of the auxiliary function value, namely the already computed value that was stored in the array entry *dataOut [table [p]]* in the previous loop as described above. Prior to these steps, the permutation array *perm* is once more randomly permuted (in the last for-loop on page 7). This procedure ensures that the order in which the steps in the following loop are executed again is different from the previous order of steps. However, such as an arrangement is optional and only serves to further avoid information leakage. The value of *dataOut* is not affected by this procedural step.

Thus, based on the actual teachings of Kocher and NOT CONJECTURE, it must be concluded that Kocher specifically requires calculation of the auxiliary function value during data falsification, and that the auxiliary function values cannot be pre-stored and retrieved from a memory in the manner claimed. In particular, based on the actual teachings of Kocher, it can be seen that in order to modify the method of Kocher to obtain the claimed invention, a number of changes would have needed to be made, none of which are suggested by Cordery:

a. the ordinary artisan would have had to recognize that the method of Kocher may, at least in principle and despite explicit teachings to the contrary, be changed without any loss of security and without changing the output values by pre-computing the random bits *b and* the random permutation *perm* so that *b* and *perm* would serve as input data in addition to the actual input data *dataIn*, *dataOut*, and *table* (which would require considerable algorithmic skills not even remotely taught by Cordery);

b. the ordinary artisan would have had to further recognize that the blinding bits *b* would need to be pre-computed in safe surroundings and stored in an array of random blinding bits, for simplicity also called *b*, and that the random permutation *perm* would also have to be pre-computed in safe surroundings;

c. the ordinary artisan would have had to further recognize that the unblinding vector, stored in the vector *dataOut*, would have had to be pre-computed by applying the permutation *perm* and the permutation *table*, in that order, to the random vector *b*, *i.e.*, *dataOut [table [perm[i]] := b[i]*; and

d. the ordinary artisan would have had to provide for storing the random vector *b* representing the auxiliary data along with the unblinding vector *dataOut* representing the auxiliary function value, with the result that the main routine to compute the actual permutation of the input array *dataIn* according to the array *table* would then comprise the following blinding steps:

```
for (i=1; i<64; i++){
p=perm[i];                          //perm has already been pre-computed
temp [p] := dataIn[p] ^ b[i];              //random vector b[i] has
                                           // already been pre-computed}
```

Furthermore, even if the main routine of Kocher were modified in such a manner, and there is nothing in Cordery to suggest such a modification, the result would still not have been the claimed invention because the blinding step occurs only *after* the permutation *perm*, representing one of the one or more operations, has been applied to the input data *dataIn.* This wold be contrary to the teachings of Kocher since the execution of the permutation *perm* before blinding serves the security purpose of randomizing the order in which the blinding steps are performed. On the other hand, the pre-computed unblinding vector would then simply read *dataOut[table[i] :=b[i]* since the respective application of the permutation *perm* would also have to be omitted in order to ensure a correct un-blinding step, resulting in a contradiction that renders the propose

modification of Kocher inoperative. **This is not a matter of conjecture, but rather follows logically from the teachings of Kocher.** On the other hand, the Examiner's conclusion that the ordinary artisan would have ignored these teachings, or considered to be non-essential and modifiable, is pure conjecture.

It is respectfully submitted that the issue that must be determined in this case is not, as alleged by the Examiner, whether Cordery has anything to do with protection of secret data (it does) or whether Cordery's data can actually be used in Kocher's device (it clearly cannot, but bodily incorporation is not the test for obviousness), but rather whether the teachings of Cordery, considered as a whole, would have led one of ordinary skill in the art to the claimed invention. In this case, the issue is:

- **whether Cordery would have led one of ordinary skill in the art to modify Kocher's method by replacing contemporaneous auxiliary function value generation with predetermined auxiliary function values and retrieval of the auxiliary values from a memory before beginning data falsification**?

Based on the above consideration of the actual teachings of Kocher, this issue must be answered in the negative. Cordery does not teach that there is a risk to generating auxiliary function values for use in disguising data operations performed by a processor. The processor of Cordery is a secure processor. All that Cordery teaches is a particularly secure way of storing predetermined sensitive data, by using an SPSD. Kocher also uses predetermined sensitive data, but how that predetermined sensitive data is determined has no effect on disguising of processor operations. In fact, Kocher is not concerned with the protection of pre-stored data. Prestored data cannot be compromised by power data analysis. Only actual processing operations generate power emissions that can be analyzed to discover secret data.

Both Kocher and the present invention are concerned with discovery of data protection algorithms, and the secret data used therein, by analyzing power emissions resulting from operation of a processor in an insecure environment. Kocher teaches one way to do so, namely by generating auxiliary function values during data falsification, and using the auxiliary function values to disguise the operations being performed as well as the secret data used in the operations. The claimed invention proposed a different way to disguise the data operations, by predetermining the auxiliary function values. This solves a problem that Kocher clearly did not

consider, namely that the same analysis methods used to compromise the operations might, in a more sensitive form, be used to discover the auxiliary values generated as part of the operations. Pre-stored auxiliary function values cannot be analyzed in this manner. The mere fact that pre-determination and storage of secret values is known, which is all that Cordery brings to the table, does not suggest modification of Kocher's method by replacing the auxiliary function value generation step with retrieval of the auxiliary function values from a memory where the auxiliary function values are stored with values that are used to falsify the data. Kocher already knows that secret data needs to be kept secret and that predetermined data needs to be stored (which is why memory 290 is provide to hold the "key"). The fact that Cordery stores the values on a removable token, and uses a secure processor to process the values, is not logically suggestive of modifying the processing algorithm of Kocher to eliminate auxiliary value generation. What possible advantage could such elimination have, other than protection against discovery by statistical analysis, which is not recognized as a problem in Kocher (with respect to the auxiliary function values), and which is clearly not a problem in the secure system of Cordery.

Cordery teaches nothing more than provision of secure data on an SPSD 104, and carrying out of processing operations using the secure data in a "secure co-processor." The operations are not carried out in an insecure environment as in Kocher, and there is no need for auxiliary functions values. If there were a need for auxiliary function values, none of the teachings of Cordery would prevent the ordinary artisan from generating the auxiliary function values during data falsification rather than pre-computing the auxiliary function values. In other words, Cordery teaches storage of sensitive pre-determined data on a removable token. It does not teach that a particular item of sensitive data required by Kocher to be generated during a specific processing operation should be replace by pre-stored data.

The failure of Cordery to suggest any sort of operation-falsification auxiliary function value storage logically follows from the fact that Cordery does not teach, or in way require, any sort of processor operation-falsification. In contrast to the setting of the present invention (and of Kocher) in which the secret data to be protected by falsifying some input data are stored on a data carrier, Cordery's secret data to be protected (in the form of decryption algorithms and keys), ***is neither stored nor executed on the data carrier 104 of Cordery but rather on a secure***

co-processor 502 separate from the data carrier 104, and which is further protected by a tamper resistant housing 513 (see col. 9, line 66 to col. 10, line 61 and Fig. 5 of Cordery). In other words, Cordery teaches protection of secret data by placing it in a separate tamper resistant, secure co-processor, which is completely contrary to the present invention, in which secret data on a chip is protected by falsifying operations on the chip and not by adding an additional secure, tamper resistant chip. A major purpose of the method of Kocher, which is part of the teachings of Kocher "as a whole," is to perform computations in an insecure environment, without the need to perform the operations in a secure environment or, by implication, to add a secure co-processor such as the one provided by Cordery.

In conclusion, the Kocher publication specifically teaches, in paragraph [0071], lines 9-12, a method in which "*The bit order table is created n two passes, where the first assures that the table has the correct form. . .and the second introduces random order into the table"* and further that: *" Because the process of constructing the bit order table does not involve any secret inputs, the only security requirement for the process is that the final result be unknown to attackers*." This method is fundamentally different than that of the claimed invention, and cannot be modified to involve the claimed pre-computation without ignoring the principles explicitly taught by Kocher, while making fundamental changes to the disclosed method. Since Cordery only teaches pre-computation in the context of a secure co-processor and without storage of compensating data, Cordery does not overcome the contrary teachings of Kocher, and withdrawal of the rejection of claims 26-33 and 42 under 35 USC §103(a) is again requested.

Having thus overcome each of the rejections made in the Official Action, expedited passage of the application to issue is requested.

Respectfully submitted,

BACON & THOMAS, PLLC

/Benjamin E. Urcia/

Date: October 29, 2010 By: BENJAMIN E. URCIA
Registration No. 33,805

BACON & THOMAS
625 Slaters Lane, 4th Floor
Alexandria, Virginia 22314
Telephone: (703) 683-0500